

実習 2 : LED制御処理 (危険領域) を排他制御 (1/2)

■ デバイスドライバの機能仕様

- read()呼び出し→write()終了の間を実行中のタスクが1つのみになるように排他制御
- read()呼び出しの時点で、すでに上記状態のタスクが存在する場合は使用中エラー

■ ユーザプログラムの機能仕様

- read()呼び出しが使用中エラーの場合、read()が成功するまで繰り返し
- write()後に設定値を記録しておき、次read()時に読み込んだ設定値と比較、制御対象LEDの(ビット)状態が変わっていたときに警告表示をするコードを追加

実習 2 : LED制御処理 (危険領域) を排他制御 (2/2)

■ デバイスドライバの実装仕様

- int型変数を使って、read()→write()間を実行中のタスク数をカウント
- read()の処理先頭で、上記変数が0でない場合はエラー (-EBUSY) を戻す処理を追加

■ ユーザプログラムの実装仕様

- read()の戻り値が-1で、かつ、エラー番号 (errnoの値) がEBUSYのときにはread()呼び出しに再度ジャンプ

■ 動作確認

1. 任意の1LEDを100ms間隔で点滅
2. 8つのプロセスを同時に起動して、それぞれ異なるLEDを100ms間隔で点滅 (複数起動は、シェルの「&」でよい)
3. 前述動作確認を10ms, 1msの点滅間隔で10~20回実行